Biomechanical Measurement System for Application to Wheelchair Seating Systems

Undergraduate Thesis

Presented in Partial Fulfillment of the Requirements for
Graduation with Research Distinction in the
Department of Mechanical Engineering at
The Ohio State University

By

Alexander Jones

Advisors:
Sandra A. Metzler, D.Sc., P.E.
Carmen P. DiGiovine, Ph.D., ATP/SMS, RET

April 2016

1. ASSISTIVE TECHNOLOGY CENTER, 2. DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING

Abstract

People who have physical or cognitive disabilities often require wheelchairs for increased mobility. Sitting in a wheelchair for an extended time can result in pressure ulcers, spinal degeneration, and other biomechanical health issues that may impact the user's health. In order to prevent these secondary health issues, the user needs to be properly fitted in their wheelchair in order to distribute the forces acting on the person in such a manner as to minimize injury risk. However, current wheelchair fitting practices are a very manual and qualitative process. Clinicians currently make educated guesses by the patient's comfort and their general dimensions, but there is no quantitative way to validate their decisions. The purpose of this research was to create a system that can experimentally calculate the center of mass for a wheelchair user. This project was also part of a larger initiative to develop a user-friendly tool for clinicians that will facilitate and improve the wheelchair fitting process. The developed measurement device utilizes force sensing resistors on specific wheelchair locations, which includes the wheelchair seat, back, and headrest. The sensors record data that is used to calculate the center of mass of the patient in respect to the wheelchair. By knowing the center of mass of the patient, clinicians will have more data for the fitting process and can be used to assess the biomechanical effectiveness of current fitting methods. By providing quantitative feedback to the clinician the efficacy of the wheelchair fitting process can be improved and the procedure streamlined. This should result in an increased quality of life for wheelchair users and provide novel data that can be utilized to better understand the biomechanical interaction of humans and wheelchair systems.

Acknowledgements

I would like to thank several individuals that contributed to the success of this project:

Dr. Sandra Metzler

Dr. Carmen DiGiovine

Amy Koehler

Ryan Letcher

Dr. Robert Siston

Matthew Yankie

The College of Engineering

The Wexner Medical Center

Table of Contents

List of Figures

List of Tables

Chapter 1: Introduction

"In every object, there is a unique point called 'center of mass (COM)' around which the object's mass is equally distributed in all directions. In other words, mass is balanced at the COM in all directions" [1]. When referring to humans in wheelchairs, knowing the user's COM can be beneficial in fitting them accurately into a wheelchair. When standing, the feet are the base of support for the body. When an individual leans too far in any direction, they become unbalanced and can no longer stand upright. This is because their COM has shifted and is no longer positioned directly above their base of support. When seated, instead of the feet being the base of support, the pelvis is the base of support. In order to achieve the maximum stability when seated, the user's COM should be centered above the pelvis and be as low as possible [2].

(a) Balanced neutral stability **(b)** Posterior Pelvic Tilt (PPT) **(c)** Anterior Pelvic Tilt (APT)

Figure 1: Pelvic Positions **[3]**

By aligning the COM, a stable pelvic position can be achieved. Obtaining a stable pelvic position can prevent the posture from degrading and additional stresses from being added onto the patient. Without the stable pelvic position, the individual will either fall over or require an additional force to balance themselves. Many individuals will use their arms to help stabilize themselves. By using their arms, they are lessening their quality of life. The infinite uses of the hands and arms that able-bodied humans have is non-existent to an individual who needs their arms to keep them upright. By achieving this stable positon enables them the full use of their arms and hands. Additionally, altering the location of the user's COM changes the resulting force distribution acting on the individual, which can affect their overall posture and comfort. If the arms Figure 1 above shows three different pelvic positions that people's pelvises may be oriented, as a result of their overall seated posture, which can included physical limitations, but may also just be their choice or preference. Figure 1a shows the pelvis in a balanced, neutral stable position. This is considered a healthy position and posture. Because majority of the masses are concentrated above the pelvic, the base of support, the user is balanced. Figure 1b shows the pelvis rotated causing the spine to arch into posterior direction giving it the name posterior pelvic tilt (PPT). Figure 1c shows the pelvis rotated in the opposite direction as Figure 1b with the spine arching in the anterior direction giving it the name anterior pelvic tilt (APT). Because of the altered pelvic rotation and the arch in the spine, the COM of the individual is different each pelvic position. When in the individual is in the posture of either Figure 2 or 3, the COM is not positioned above the pelvis. Based on this, the force of gravity acting on the body may cause instability of person and add additional stresses on the individual's body. Because of these additional stresses, PPT and APT are positions that are generally to be avoided or corrected by clinicians [3].

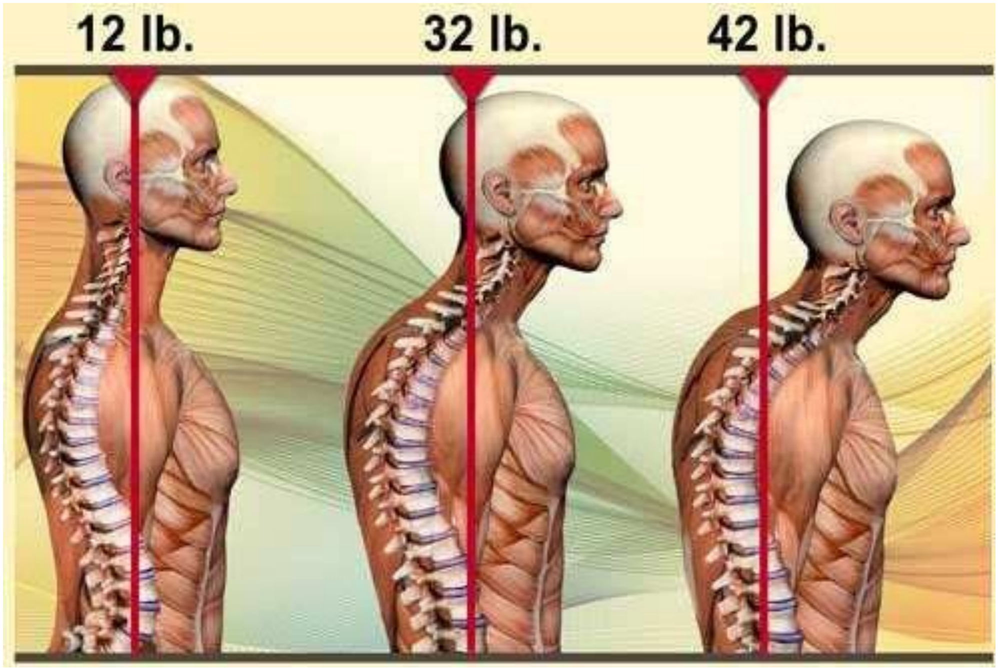

Figure 2: Relative Weight of the Head Over the Body **[4]**

As an individual's COM is shifted out of line, the head is one part of the body that can cause additional stresses on the rest of the body. Generally, the human head weighs around twelve pounds. When this weight is not directly above the pelvis, it is also not above the spine and shoulders, which are responsible for supporting the head. When the head is in a forward positon for a significant amount of time, the relative weight the head has on the cervical extensors, upper neck muscles, increases "For every 1 inch your head moves forward, the relative weight of the head over the body doubles due to the effects of gravity." Figure 2 above illustrates the relative weight of the head as it is shifted forward. This additional weight applies an abnormal amount of strain on the shoulder and neck muscles and ligaments causing the head to un-stabilized and to continue to move forward further increasing the issue. Additionally, negative effects can occur in decreased efficiency in the body's respiratory and nervous functions [4, 5].

Currently, during the wheelchair fitting process, a clinician will take the general dimensions of the patient (height, leg length, hip width, etc.) to ensure the correct sized wheelchair is selected. Additionally, a pressure mat will be utilized on various wheelchair seat cushions to measure the pressure distribution between the user and the wheelchair seat cushions. Below, Figure 3 shows part of the output from the pressure mats. The figure shows the pressure distribution on the user based on their contact with the wheelchair seat. By studying this data, clinicians can understand the where the high points of pressure are on the user. Along with aligning the COM, limiting points of high pressure are essential to a successful wheelchair fitting. The points of high pressure can cause additional health concerns that could cause a worsening posture, discomfort, pressure ulcers, and other additional health concerns.

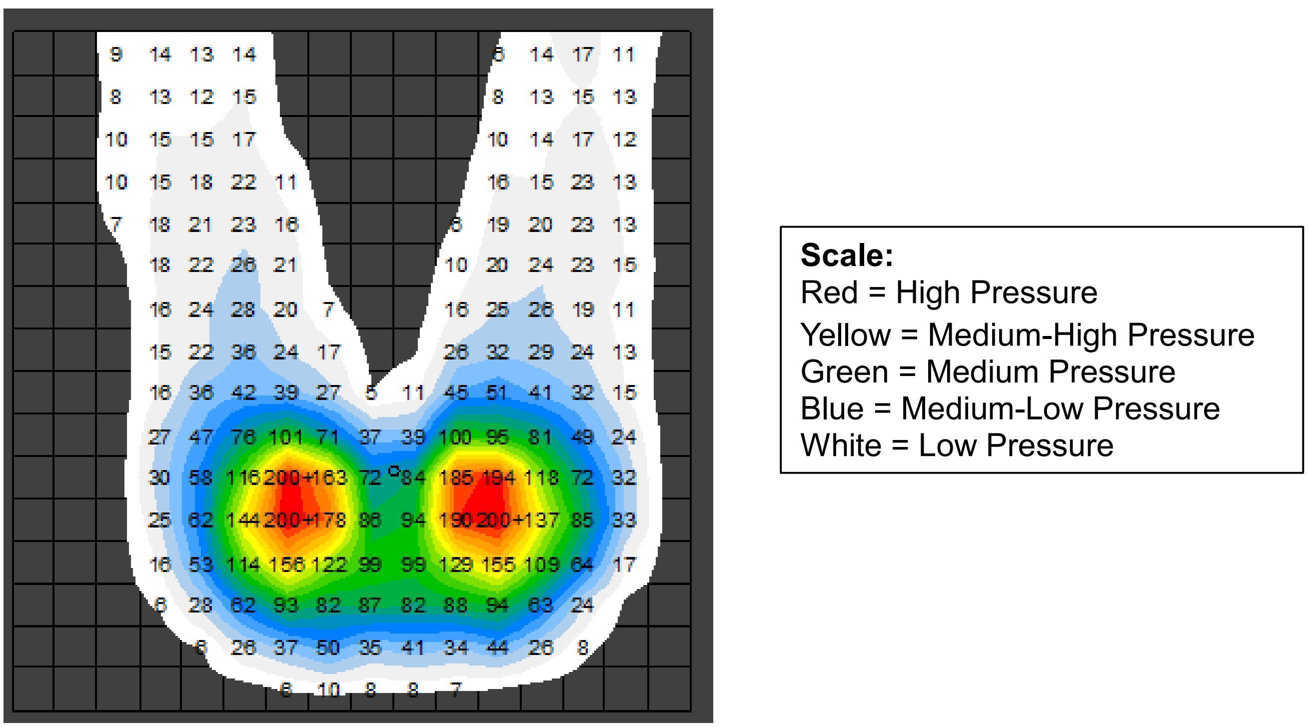

Figure 3: Pressure Mapping

Currently, biomechanics and statics principals are used as a basis for the decisions made during the wheelchair fittings. However, the fittings remain a manual and primarily qualitative process since hardly any data is being measured beyond the general body dimensions and the use of the pressure mat. The determination of what is satisfactory varies from clinician to clinician as the decisions are qualitative. Although it is understood by clinicians that COM is essential in aligning the pelvis and the individual, there exists no system to calculate or measure the patient's COM during the fitting process. It is more of an art than science. There is currently no repeatable method by which clinicians can accurately and quantitatively measure and analyze the center of mass of the patient in the wheelchair.

Because COM is known to be an essential piece of data to help effectively fit someone into a wheelchair, alongside of this project, a segmented biomechanical model is being created. By using previous studies that estimated the mass of each body segment as a percentage of total mass, this model is able to position the person in various seated orientations and theoretically calculate the COM. Additionally, it also is able to show how forces from the wheelchair and its attachments act on the body.

1.1 Focus of Thesis

The main purpose of this study is to develop a system to experimentally calculate the center of mass for wheelchair users. Utilizing the ability of force sensitive resistors to measure the force applied to them, a system was created that can measure a user's COM when they are sitting in a wheelchair. The system is powered by an Arduino Uno and controlled by MATLAB. This project is also part of a larger initiative to develop a user-friendly tool for clinicians that will facilitate and improve the wheelchair fitting process.

1.2 Significance of Research

This project creates a proof of concept prototype that is the first step in creating a system to allow clinicians the ability to calculate the COM of patients during wheelchair fittings. By developing a system will allow for the center of mass to be experimentally measured of a wheelchair user in various positons. Additionally, this project allows for the comparison of this experimental model to the biomechanical model being developed.

By having more knowledge when fitting patients, fitting patients becomes more effective, increasing their quality of life. This will add more data to the fitting process, allowing a better understanding of the effectiveness of the fitting methods. This can then help build more quantitative clinical practices. By having practices that are more quantitative, the variability between clinicians will decrease. Additionally, this device can be used by wheelchair designers and suppliers to continue to help improve the design of the wheelchair and its components.

1.3 Overview of Thesis

This thesis will present my research work on developing a measurement system for experimentally measuring COM for wheelchair users. This thesis is structured in 7 chapters. The first chapter introduces the project and provides details on the focus and the significance of the research. Chapter 2 details the requirements developed for the measurement system. Chapter 3 provides details on the sensors chosen for the device. Chapter 4 details how an Arduino board and MATLAB were utilized. Chapter 5 discusses how the location of the sensors was calculated. Chapter 6 summarizes how the design comes together in MATLAB to output the desired results. Finally, Chapter 7 concludes the thesis. Appendices can be located after the conclusion.

Chapter 2: Measurement System Requirements

2.1 Introduction

The objective of the project was to develop a system that can accurately measure the COM of an individual in a wheelchair. The desired parameters and constraints to be met were the following:

1. **Unobstructive**: The system should not interfere with the patients comfort or posture in the chair. The size of sensors and other components need to be thin or small enough to give results as if the patient were in a wheelchair with nothing in-between them and the surface of the chair.
2. **Adaptability**: The system needs to be adjustable to account for patients with differing body dimensions. The sensors need to be able to be moved around to record the appropriate data relative to each person. Additionally, the system needs to be capable of being used on various wheelchairs.
3. **Modular**: The system need to allow the clinician or researcher to add/remove sensors as needed for the individual in the chair

2.2 Finding the Total Center of Mass

Mathematically, the general term for COM is expressed as:

$$X = \frac{1}{M}\sum_{i=1}^{n} m_i x_i \qquad [1]$$

where, X, is the X coordinate of the location of the COM, M is the total mass of the body, m_i represents the masses of each segment or particle of the body, and x_i is the x location of the masses of each segment [6]. Applying this to this project, the two main parameters needed to solve equation 1 and to find the person's total COM is (1) the mass of each segment of the body, and (2) the location of that mass, or the COM of each segment.

When applying this equation to humans, each body segment (i.e. head, neck, arm, leg) make up the total body. Having the mass of each segment and where each of those masses are in space are essential to find the COM. There have been several studies in the past century finding the COM of each body segment and the total COM of an average human. One study in particular, shown in Winter's *Biomechanics and Motor Control of Human Movement*, found the mass of each body segment as a percentage of the person's total mass. Additionally, the study found the average location for the COM of each segment as a percentage of the segment length. With those ratios, the total COM could be calculated for any individual if their body was proportionate to the study's "average" person. Table 1 below shows this study's findings [6].

Table 1: Study that calculated the mass each body segment as a percent of total mass as well the location of COM for each segment with respect to the length of the segment **[6]**.

Segment	Definition	Segment Weight/Total Body Weight	Center of Mass/Segment Length Proximal	Center of Mass/Segment Length Distal	Radius of Gyration/Segment Length C of G	Radius of Gyration/Segment Length Proximal	Radius of Gyration/Segment Length Distal	Density
Hand	Wrist axis/knuckle II middle finger	0.006 M	0.506	0.494 P	0.297	0.587	0.577 M	1.16
Forearm	Elbow axis/ulnar styloid	0.016 M	0.430	0.570 P	0.303	0.526	0.647 M	1.13
Upper arm	Glenohumeral axis/elbow axis	0.028 M	0.436	0.564 P	0.322	0.542	0.645 M	1.07
Forearm and hand	Elbow axis/ulnar styloid	0.022 M	0.682	0.318 P	0.468	0.827	0.565 P	1.14
Total arm	Glenohumeral joint/ulnar styloid	0.050 M	0.530	0.470 P	0.368	0.645	0.596 P	1.11
Foot	Lateral malleolus/head metatarsal II	0.0145 M	0.50	0.50 P	0.475	0.690	0.690 P	1.10
Leg	Femoral condyles/medial malleolus	0.0465 M	0.433	0.567 P	0.302	0.528	0.643 M	1.09
Thigh	Greater trochanter/femoral condyles	0.100 M	0.433	0.567 P	0.323	0.540	0.653 M	1.05
Foot and leg	Femoral condyles/medial malleolus	0.061 M	0.606	0.394 P	0.416	0.735	0.572 P	1.09
Total leg	Greater trochanter/medial malleolus	0.161 M	0.447	0.553 P	0.326	0.560	0.650 P	1.06
Head and neck	C7–T1 and 1st rib/ear canal	0.081 M	1.000	— PC	0.495	0.116	— PC	1.11
Shoulder mass	Sternoclavicular joint/glenohumeral axis	—	0.712	0.288				1.04
Thorax	C7–T1/T12–L1 and diaphragm*	0.216 PC	0.82	0.18	—	—	—	0.92
Abdomen	T12–L1/L4–L5*	0.139 LC	0.44	0.56	—	—	—	—
Pelvis	L4–L5/greater trochanter*	0.142 LC	0.105	0.895	—	—	—	—
Thorax and abdomen	C7–T1/L4–L5*	0.355 LC	0.63	0.37	—	—	—	—
Abdomen and pelvis	T12–L1/greater trochanter*	0.281 PC	0.27	0.73	—	—	—	1.01
Trunk	Greater trochanter/glenohumeral joint*	0.497 M	0.50	0.50	—	—	—	1.03
Trunk head neck	Greater trochanter/glenohumeral joint*	0.578 MC	0.66	0.34 P	0.503	0.830	0.607 M	—
Head, arms, and trunk (HAT)	Greater trochanter/glenohumeral joint*	0.678 MC	0.626	0.374 PC	0.496	0.798	0.621 PC	—
HAT	Greater trochanter/mid rib	0.678	1.142	—	0.903	1.456	—	—

*NOTE: These segments are presented relative to the length between the greater trochanter and the glenohumeral joint.
Source Codes: M, Dempster via Miller and Nelson; *Biomechanics of Sport*, Lea and Febiger, Philadelphia, 1973. P, Dempster via Plagenhoef; *Patterns of Human Motion*, Prentice-Hall, Inc. Englewood Cliffs, NJ, 1971. L, Dempster via Plagenhoef from living subjects; *Patterns of Human Motion*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1971. C, Calculated.

Chapter 3: Sensors

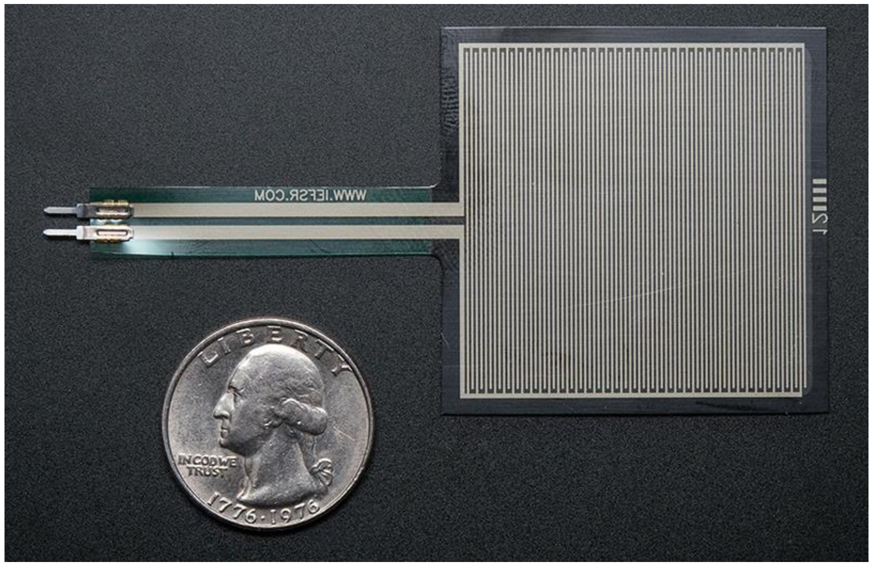

Figure 4: FSR Sensor **[7]**

In order to capture the mass of each segment, one of the two desired parameters of the COM equation, this project utilizes several Force Sensing Resistor (FSR). Thick, bulky force sensors or transducers placed under an individual would be uncomfortable to the individual as well as add additional factors that could alter the accuracy of the data recorded. Because of this, the FSR is a good choice based on how thin and flexible it is. Additionally, the FSR is an inexpensive sensor (around $7 each) allowing the large population of people and institutions to be able to purchase one or several sensors [7].

As force is applied, the FSR changes its resistance. The more force applied, the less resistance from the FSR. Figure 5 below illustrates this relationship.

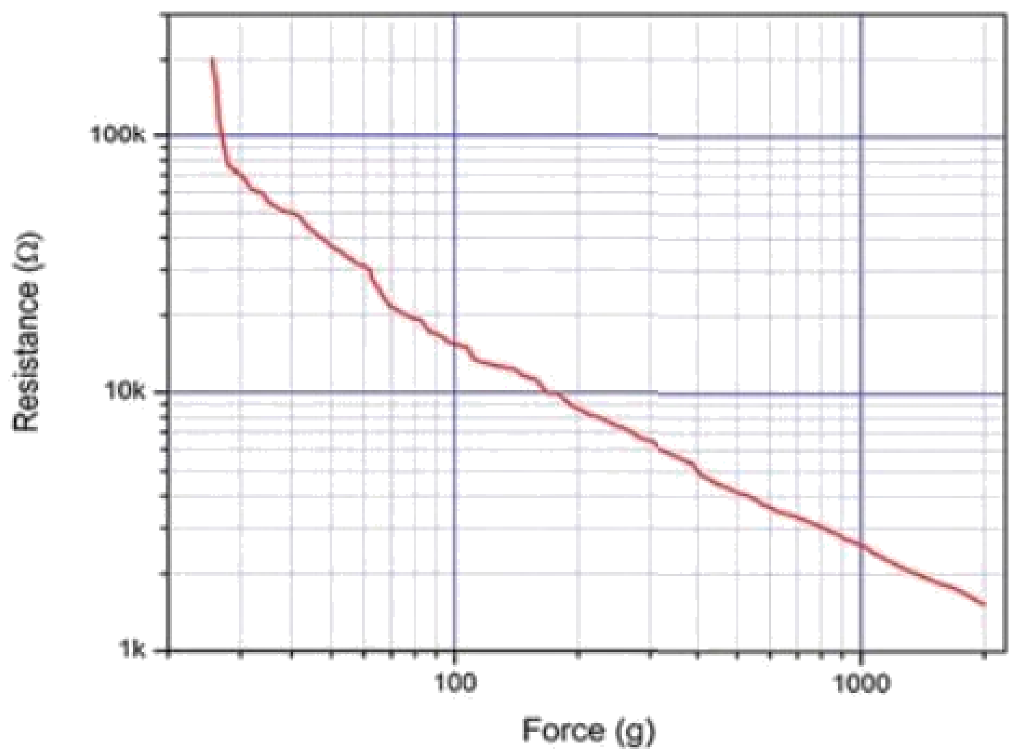

Figure 5: Relationship between force and resistance in FSR sensors **[8]**

To achieve the relationship between forces applied to voltage output for each sensor, the sensors are set up in a voltage divider configuration that can enable the FSR to perform a simple force-to-voltage conversion. The output to this setup can be described with the following equation:

$$V_{out} = \frac{V^{+}}{\left[1 + \frac{R_{FSR}}{R_M}\right]} \qquad [2]$$

where, V_{out} is the voltage outputted from the FSR, V^{+} is the voltage inputted to the FSR, R_{FSR} is the resistance of FSR that is changing based on the force applied to it, and R_M is an additional resistor which effects the force sensitivity range [8]. V_{out} is the parameter that is recorded and converted into force and COM. Figure 6 and Figure 7 shows how the FSR was setup electronically to achieve the voltage divider configuration. This configuration in Figure 7 was repeated for each sensor, using a different analog in port for each sensor.

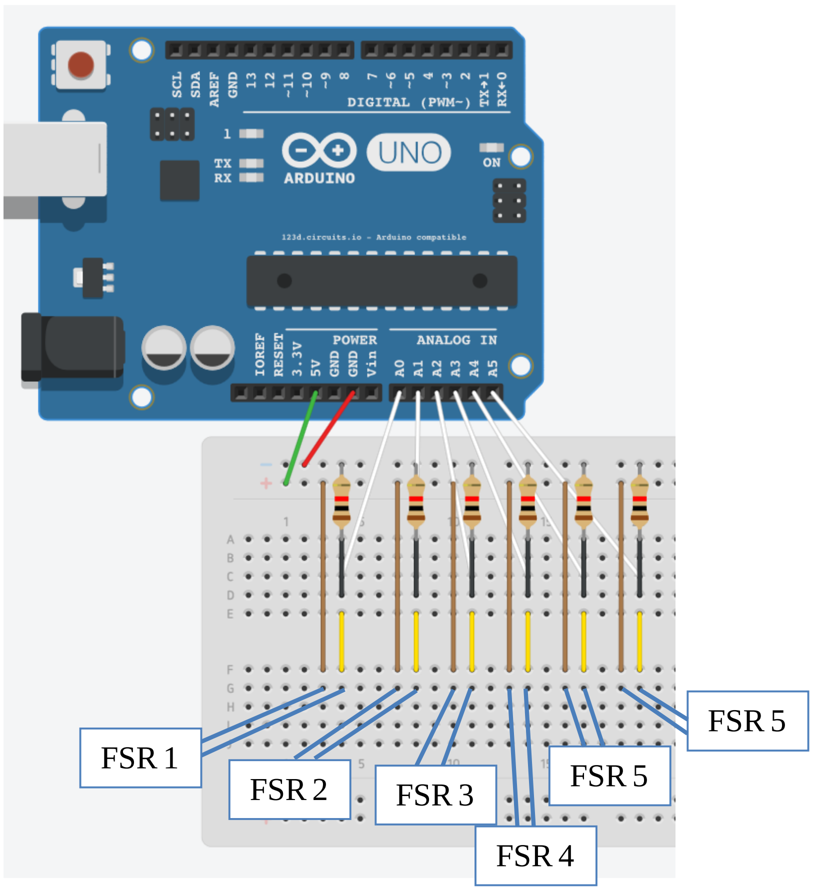

Figure 6: Circuit Configuration

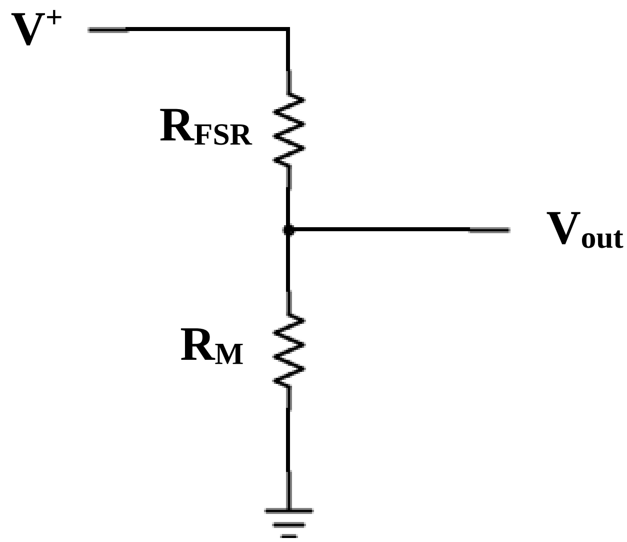

Figure 7: Circuit Schematic for one sensor

There is a logarithmic relationship between the output voltage and the force applied to the sensors. As the force applied increases, the voltage difference between readings is decreases. This poses the issue of accuracy. If the voltage that corresponds to force A is very close to the voltage corresponding to force B, then the system will not accurately be able to calculate if the force applied when it is a force close to magnitude to force A and B.

One method of increasing the voltage difference between readings is by altering resistor R_M, in the voltage divider configuration. By changing R_M, changes the force sensitivity range of the sensor. As the resistor's value is decreased, the range the sensors can detect is increased. Figure 8 below illustrates this behavior for various resistors.

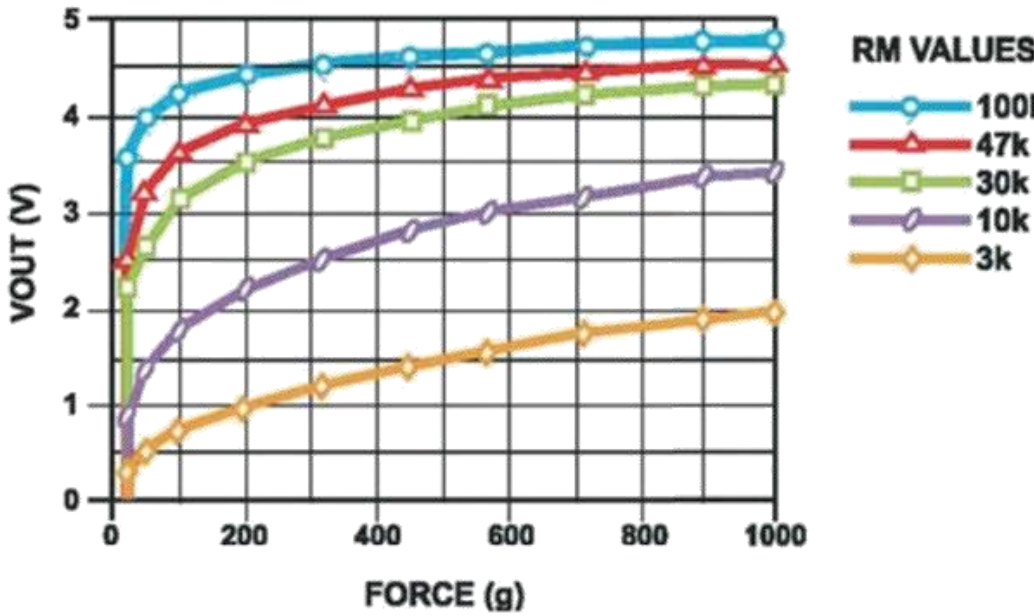

Figure 8: Force vs. Voltage with Different R_M Values **[8]**

The needs of this project entail a large percentage of a human's total weight to be on the sensor. When sensors are placed under a person's ischial tuberosities (ITs), the curved bones forming the base of the pelvis, the force can be, close to the weight of the person's trunk, arms, and head (HAT). Using the study displayed in Table 1, the weight of the individual's HAT is 0.678 of their total weight. Using a person who weighs 172 lbs. based on the average human weight used in the theoretical model study being done alongside this one, both ITs need to be able to capture up to 117 lbs. or around 58 lbs. each [6]. This value is assumed to be the max value. (Note: This value represents when the user is exclusively making contact with their ITs, this values is impossible to reach, but is sufficient for using as a max in this instance) By using that as a max value, a 1k ohm resistor is sufficient to use for reading values that reach around 60 lbs.

The conversion from the output voltage to force is achieved during the calibration of each sensor. The sensors were calibrated by recording the voltage values when known incremented masses were placed on the sensors. A logarithmic calibration curve was applied to the data when plotted force vs voltage. Figure 9 shows one of the sensor's calibration curves. The remaining calibration curves can be found in Appendix A.

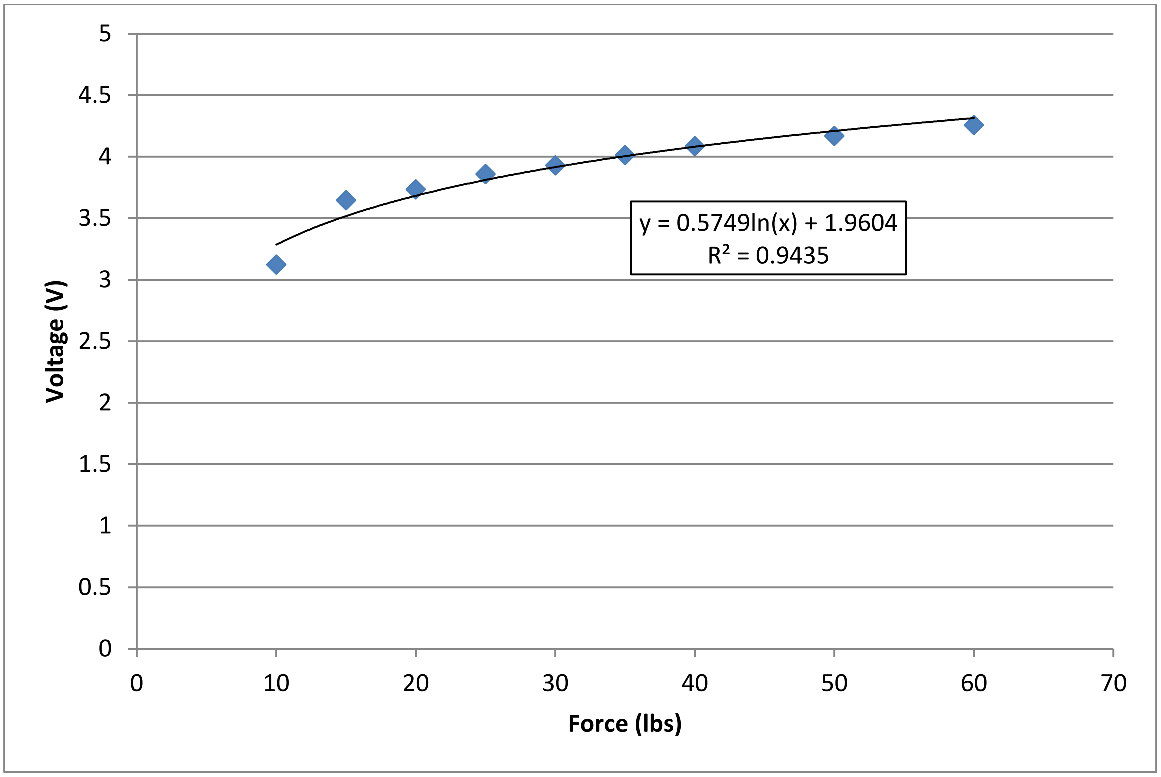

Figure 9: Sensor 1 Calibration Curve

Chapter 4: Arduino and MATLAB

4.1 Arduino Uno

Figure 10: Arduino Uno Microcontroller **[9]**

The FSR sensors is powered and controlled by two Arduino Unos, displayed in Figure 10 above. Arduino is an "open source electronics platform based on easy-to-use hardware and software." There are several Arduino boards available. The most common, the Arduino Uno was chosen. The Arduino was chosen because it is inexpensive ($24.50), can be programmed using the provided software or MATLAB. It is also open source which allows for external tutorials and instructions on a variety of capabilities to be readily available [9].

Additionally, compared to some other devices, the Arduino Uno offers six analog input ports. Analog allows for the voltages and the data to remain closer to how they actually exist. Analog signals are those that can vary within a range. Theoretically they can be an infinite number of values. The opposite, discrete ports, would only allow the data at certain values across the same range. Therefore, because discrete data can only be a finite amount of values, it is not as accurate compared to analog [10]. Because more than six sensors will be used for this system, two Arduino Unos were used to allow up to 12 analog pins to be used. The Arduino Uno also can output 5V unlike other boards only capable of 3.3V. This is helpful when converting from voltage to force. By going up to 5V, the voltage differences between force values will be higher than the 3.3V allowing for more accuracy [9].

4.2 MATLAB

MATLAB, a well-known platform “optimized for solving engineering and scientific problems,” is a program often used to express computational mathematics and to visualize data [11]. MATLAB is often used in mechanical engineering courses at The Ohio State University, providing me with prior knowledge of how to use the program. Recently, a MATLAB Support Package for Arduino Hardware was added. This support package enables MATLAB to send commands to the Arduino and receive data through a USB cable [12]. This feature is useful for this project because it prevents the need to use the Arduino provided program to control the Arduino boards. Additionally, MATLAB simplified the commands needed to send commands to the Arduino board making MATLAB as the primary program to control the sensors and run mathematical operations on the data.   The MATLAB code used for this project can be found in Appendix B.

Chapter 5: Location of Sensors

5.1 Placement of Sensors

Up to 10 FSR sensors will be used in the system. They are placed on wheelchair where the wheelchair user comes into contact with the wheelchair. The total COM of a person is calculated by having all of the individuals mass recorded [6]. With this system, in order to have an accurate COM reading of the person, the closer to the total mass of the individual that is detected by the sensors, the more accurate. In order to capture most mass per segment, the sensors are placed where each segment's COM is estimated, shown in Table 1, or where peak forces are present from the person onto the chair. Many of the segments in Table 1, such as the abdomen or hand are difficult to measure because the segment does not come directly into contact with the chair. This eliminates a surface for the sensor to be attached to gain readings. To solve this, several segments are combined and treated as one joined segment. The combined mass and COM to be imported into Equation 1.

To simplify the system, the person is modeled as three segments. The first segment is of the lower legs and feet. This is measured by a sensor under each heel. The second segment is of the upper leg, or thighs. Two sensors are used to measure this, each placed under the user's thigh. The location of the sensor under the thigh is, as previously described, uses the location of COM of the thigh calculated in Table 1. The table shows both the proximal and distal locations of the COM as a percentage of the total segment. For the thigh segment, the below equation using the Table value of 56.7% of segment length illustrate how the location is found:

$$Location\ of\ \ Sensor = 0.567 \times thigh\ length \qquad [3]$$

The third segment is comprised of the head, arms, and trunk (HAT). Two sensors will be placed under the user's ITs accounting for the user's pelvis, two sensors are placed under the user's PSIS accounting for the trunk (minus the pelvis), and two sensors behind the peak of the back or shoulder blades accounting for the user's head, neck and arms. These locations were chosen because they are generally easy locatable landmarks and have more contact force between the user and the wheelchair compared to other parts on the users back.

When a person is seated, the weight of their body is distributed over the surface that is supporting them, which can include the seat, the backrest, or the footrest. Although the sensors are placed where there is a larger concentration of mass on segments, there is a remaining amount of mass not being captured by the sensors. In order to account for this missing mass, a correction factor is created and applied to overcome this limitation.

$$Correction\ Factor = \frac{Theoretical\ Value}{Recorded\ Value} \qquad [4]$$

Using the percent of total mass that theoretically should be present at the COM for each segment from Table 1, the conversion factor is found by dividing recorded sensor values from the theoretical values. Equation 3 above illustrates this. For the foot and lower leg segment, the theoretical value is 6% of the total user's mass (accounting for both feet and lower legs). For the thigh segment, the theoretical value is 10% of the user's mass (accounting for thighs). For the HAT segment, the ITs, PSIS and peak of the back will not have the same correction factor based on the uneven distribution of mass throughout. The theoretical values also use the percent of total mass from Table 1. Additionally, with this segment, the angle of the seatback can affect the magnitude of values recorded by the sensors, affecting the correction factor. The free-body diagram shown in Figure 11 and equations below illustrate how the theoretical forces for correction factor is calculated for this segment.

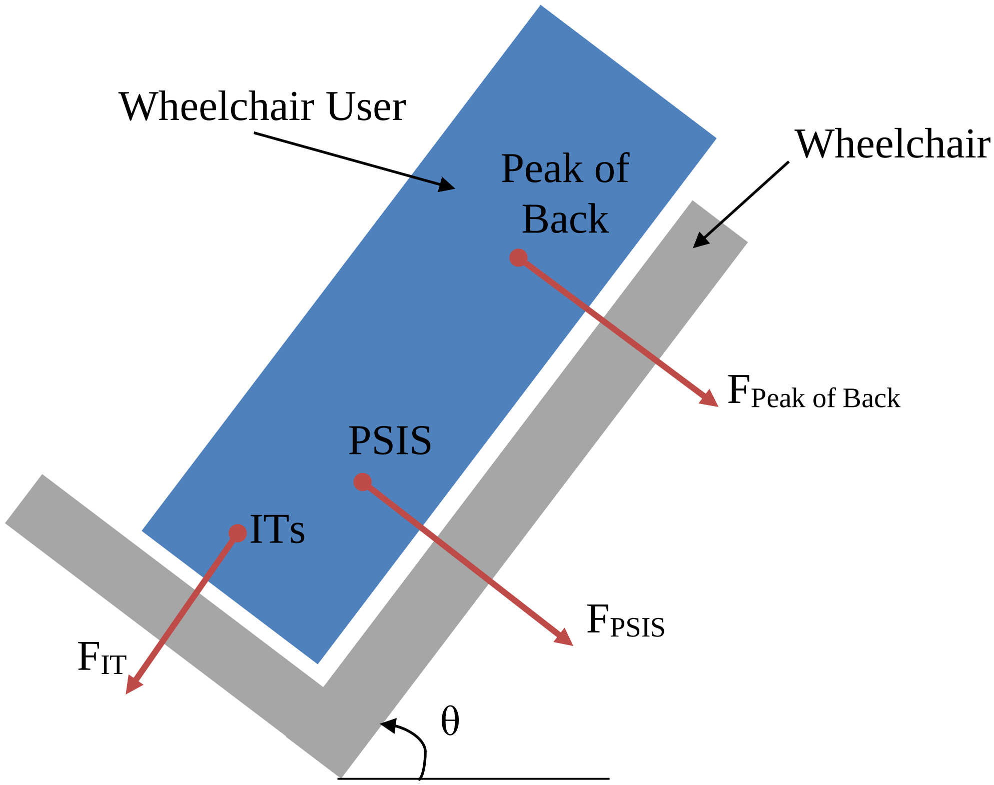

Figure 11: Free Body Diagram of Side view of HAT Segment

$$F_{IT} = m_{IT} g \times \cos\theta + m_{PSIS} g \times \sin\theta + m_{Peak\ of\ Back} g \times \sin\theta \quad [5]$$

$$F_{PSIS} = m_{PSIS} g \times \cos\theta \quad [6]$$

$$F_{Peak\ of\ Back} = m_{Peak\ of\ Back} g \times \cos\theta \quad [7]$$

$$g = 32.2\ ft / s^2 \quad [8]$$

Segment masses from Table 1:

$$m_{IT} = 0.142 \times M_{Total} \quad [9]$$

$$m_{PSIS} = 0.355 \times M_{Total} \quad [10]$$

$$m_{Peak\ of\ Back} = 0.181 \times M_{Total} \quad [11]$$

$$M_{Total} = User's\ Total\ Mass \quad [12]$$

The calculations below show how the theoretical values is found for each sensors in order to calculate the correction factor in Equation 4. For each equation, constants from Table 1 are used with the wheelchair user's total weight.

For each foot sensor:

$$Foot\ and\ Leg\ Weight = \frac{0.061}{2} \times User's\ Total\ Weight \qquad [13]$$

For each thigh sensor:

$$Thigh\ Weight = \frac{0.100}{2} \times User's\ Total\ Weight \qquad [14]$$

For each IT sensor:

$$IT\ Weight = \left(\frac{0.142}{2} \cos(\theta) + \frac{0.181}{2} \times \sin(\theta) + \frac{0.355}{2} \times \sin(\theta) \right) \times User's\ Total\ Weight \qquad [15]$$

For each PSIS sensor:

$$PSIS\ Force = \frac{0.355}{2} \times \cos(\theta) \times User's\ Total\ Weight \qquad [16]$$

For each sensor under the peak of the back:

$$Peak\ of\ Back\ Force = \frac{0.181}{2} \times \cos(\theta) \times User's\ Total\ Weight \qquad [17]$$

The following table shows the measured and theoretical force values each sensor recorded and the corresponding correction values. These calculations were done with a user's total weight equal to 125 lbs. and the chair at 10 degrees.

Table 2: Correction Factor Calculations

	Sensor	Measured Value	Theoretical Value	Correction Factor
IT	1	17.84752	14.55738276	0.815652974
	2	13.45938	14.55738276	1.081579
Thigh	3	8.4876	6.25	0.736368349
	4	9.52044	6.25	0.656482263
PSIS	5	1.36995	21.85042202	15.94979526
	6	2.534017	21.85042202	8.622839555
Peak of Back	7	2.111567	11.14063771	5.276004837
	8	3.2223	11.14063771	3.457355835
Foot	9	19.9	3.75	0.188442211
	10	24.324	3.75	0.154168722

5.2 Attachment to the Wheelchair

A cloth is pinned overtop of the seat and another piece of cloth over the back of the wheelchair, where the user will sit or lean against. Velcro strips are sewn onto the cloth. The complementary strip of Velcro is stuck to backside of each sensor. The strips on the cloth are in the general area that the locations of the sensors are expected to be. The area is enlarged to allow for a variety of sizes individuals and for adjustability in order place sensors in correct location.

The pieces of cloth are also gridded in one inch increments. This grid is to allow for the user to know the location of the sensors, which is the second desired parameter to obtain the COM through Equation 1. Because the grid is two-dimensional and the COM needs to be in a three-dimensional space, the seat height, and angle of the backrest, both manually measured, will be used to gain all x, y, and z coordinates for each sensor. Figure 12 below shows an image of the system setup with the gridded cloths.

Figure 12: Wheelchair Setup

The Figures 13 and 14 show the top and front views of the wheelchair to illustrate how the setup of the sensors is done on the wheelchair seat and back rest with explanation for each distance. Many of the sensors are determined by feeling the user's ITs, PSIS, or peak of back and adjusting the sensors manually to line up with the landmarks. The origin for of the grid on the wheelchair seat is the back left corner of the chair when facing the chair head on. The origin for the grid on seatback of the chair is in the bottom left corner of the grid when facing the wheelchair head on. The location of the sensors under the user's heels is determined by manually measuring the x and y distances from the origin on the wheelchair seat. Its z distance, is measured from the ground.

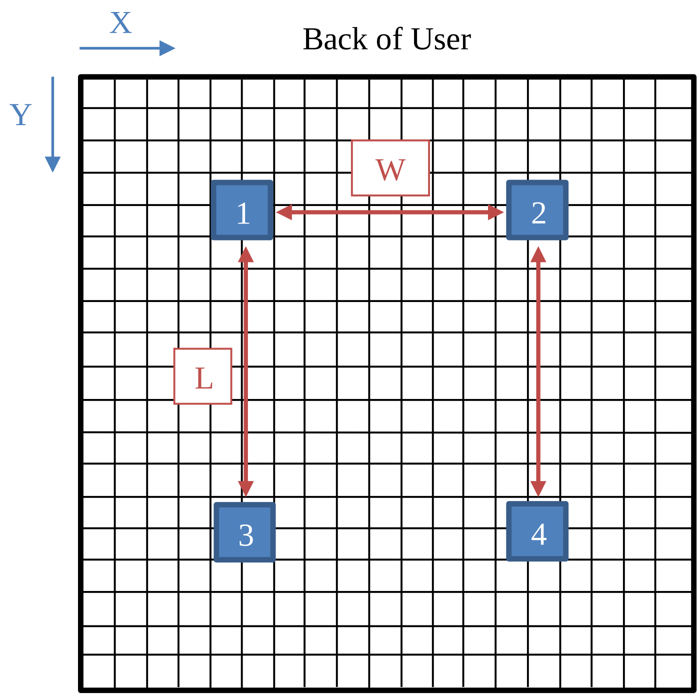

Figure 13: Location of Sensors on Wheelchair Seat (Top View)

$$1\&2 = IT\ Sensors$$
$$3\&4 = Thigh\ Sensors$$
$$W = Determined\ by\ the\ location\ of\ the\ User's\ ITs$$
$$L = Determined\ through\ equation\ 3$$

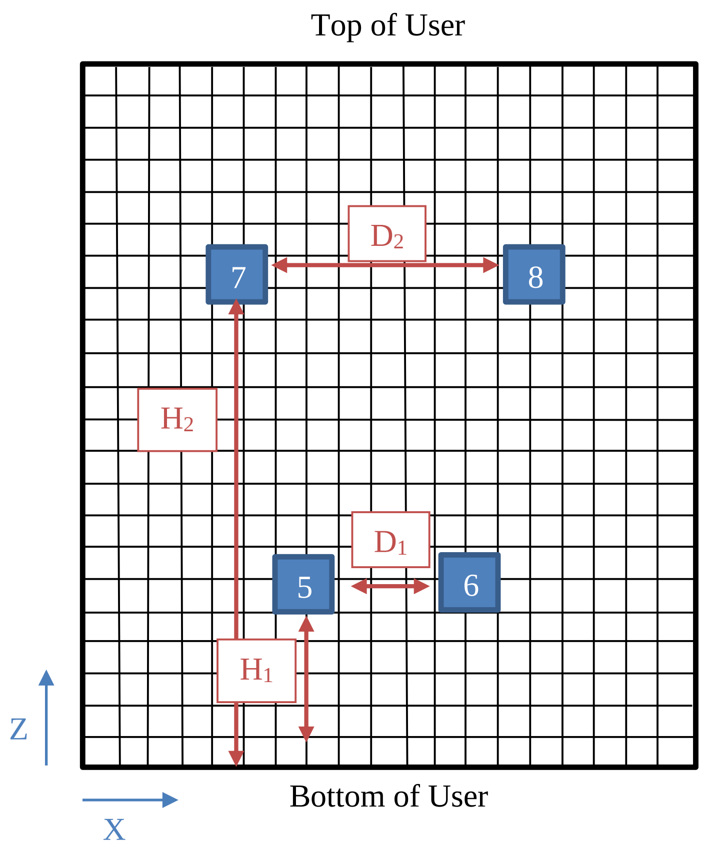

Figure 14: Location of Sensors on Wheelchair Backrest (Front View)

$$5\,\&\,6 = PSIS\ \ Sensors$$

$$7\,\&\,8 = Peak\ of\ \ Back\ Sensors$$

$$H_1\ \&\ D_1 = Determined\ \ by\ \ the\ \ location\ \ of\ \ the\ \ User's\ \ PSIS$$

$$H_2\ \&\ D_2 = Determined\ by\ the\ location\ of\ \ the\ User's\ Peak\ of\ \ Back$$

Chapter 6: Calculating COM through MATLAB

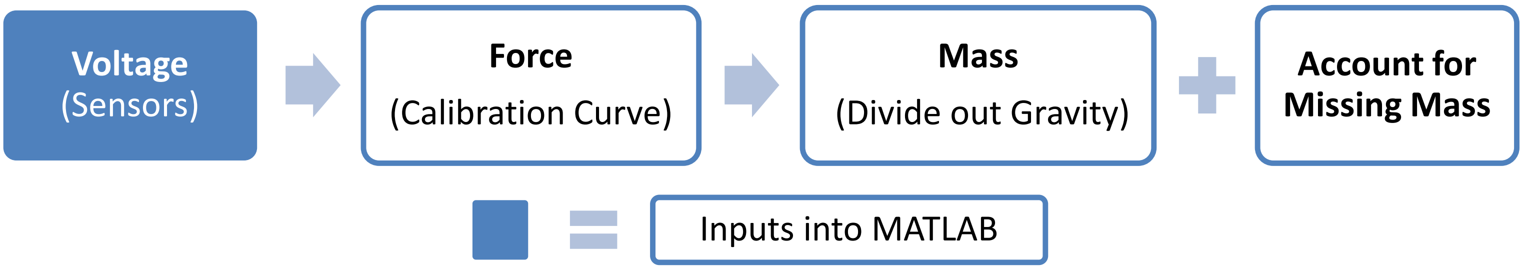

Figure 15: Determining mass with sensors

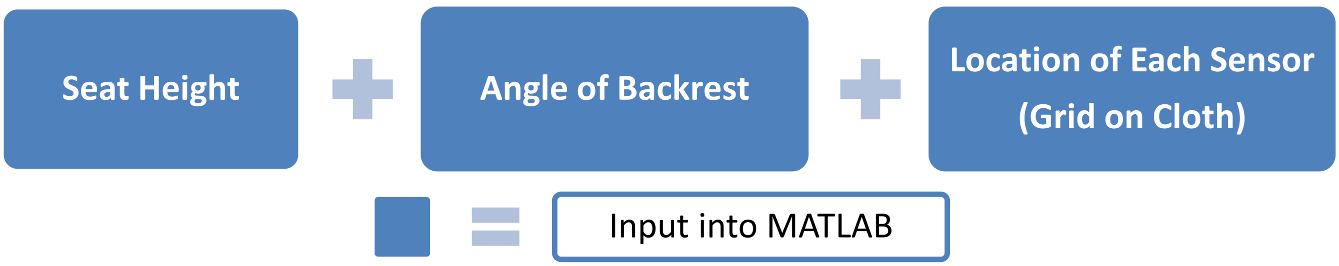

Figure 16: Determining Location of Sensors

Figures 15 and 16 above illustrate a simplified version of how the system and MATLAB work to provide the two parameters needed to solve for the total COM in Equation 1, mass and location of each segment. The boxes in each figure that are blue are parameters that are input into the MATLAB script each time it is run.

With determining the mass as shown in Figure 15, force is applied onto the sensors from the individual in contact with the sensors, resulting in an output voltage recorded by MATLAB. MATLAB then converts the voltage data into force using the calibration curve calculated, specific to the sensor the voltage is from. Using Newton's second law of motion, the acceleration from gravity can be divided out resulting in mass. Because gravity acts only in the vertical direction, trigonometry will be used with the backrest angle to calculate the amount of gravitation is acting on the sensors when reclined. Lastly, the correction factor to make up for the mass not being sensed by the sensors is applied to the mass recorded. Then the system will have a value for m_i in Equation 1 for each sensor.

For determining the location of the mass, as illustrated in Figure 16, all parameters are inputted manually into MATLAB script. The script prompts the operator of the MATLAB code for the seat height and the angle of backrest. Both of these are measurements often measured by clinicians during fitting procedures. Additionally, the code asks for location of each sensor. This is the data determined from the gridded cloth pinned to the chair seat and backrest from the grid on the cloth will be manually entered, when prompted by MATLAB. These inputs are used to determine the x, y, and z locations (with respect to the pinned cloth on the wheelchair) that will be entered into Equation 1.

Below shows step by step how the system works theoretically:

1. The sensors need to be placed based on the wheelchair user's dimensions. The locations of the sensors on the seat or backrest need to be recorded using the gridded cloth.
    a. The sensors on the back of the chair are measured based on the locations of the user's PSIS and peak of back. These are adjusted by feeling the users back to ensure the correct landmarks rest center on the sensor.
    b. Similarly, the sensors on the ITs are found by feeling the user's ITs and aligning the sensors accordingly on the wheelchair seat
    c. The upper leg, or thigh, sensors are placed at the COM location of the user's thigh. The length of the thigh is measured and multiplied by 0.567 as shown in Equation 3. The resulting value is the distance of the thigh COM location specific for this user, measured distally, or from the pelvis towards the feet.

   d. The location for the sensors under the feet is under the user's heel where majority of the force from the feet is located.

2. Next, the following needs to be inputted into MATLAB when prompted:
   a. Wheelchair user's total weight
   b. The wheelchair recline angle
   c. The distance from the ground to the top of seat cushion
   d. The distance from the ground to the bottom of the backrest
   e. The distance from the ground to the top of the foot rest
   f. The coordinates of each sensor from the gridded cloths (X & Y for the seat cushion, X and Z for the seat back)
3. Afterwards, the MATLAB code performs the following equations (user input is not required for these steps):
   a. The Y dimension of the backrest sensors is calculated using the wheelchair recline angle.

$$Z_{PSIS} = Y_{PSIS} \times \cos(90-\theta) \quad [18]$$

$$Z_{Peak\ of\ Back} = Y_{Peak\ of\ Back} \times \cos(90-\theta) \quad [19]$$

$$\theta = Wheelchair\ recline\ angle \quad [20]$$

4. MATLAB then reads the voltage output from each sensor and relates the voltages to force using the calibration equations found during the calibration of each sensor
   a. The logarithmic equations from each calibration chart is solved for X. Y represents the voltage output and X represents the force value
   b. The voltage reading is inserted into the equation as the Y value. This outputs an X value that represents force.

5. Next, MATLAB divides out the gravitational acceleration to get values of mass for each sensor.

   a. Because gravity is acting straight down, the angle of the seatback alters the amount of gravitational acceleration acting on the PSIS and Peak of Back sensors. The following equations are used for finding the mass values for those sensors.

$$m_{PSIS} = \frac{F_{PSIS}}{g * \sin(90 - \theta)} \qquad [21]$$

$$m_{Peak\ of\ Back} = \frac{F_{IT}}{g * \sin(90 - \theta)} \qquad [22]$$

$$g = gravitational\ acceleration = 32.2 \frac{ft}{s^2} \qquad [23]$$

   b. The remaining sensors that are not located on the seatback receive the total gravitation acceleration and are calculated using the following equation

$$m = \frac{F}{g} \qquad [24]$$

6. The correction factors previously mentioned are then applied to the mass values. The recorded mass value is simply multiplied by the corresponding correction value result in a more accurate mass value. Equation 25 below illustrates this.

$$m_{corrected} = m_{measured} \times Correction\ Factor \qquad [25]$$

7. At this point, all the required values needed for the total body COM calculations are known. MATLAB then takes these values and enters them into the following equations. The output from these equations are the X, Y, and Z locations of the user's total body COM in relation to the gridded cloth on the wheelchair.

$$COM_x = \frac{m_{IT1}x_{IT1} + m_{IT2}x_{IT2} + m_{PSIS1}x_{PSIS1} + m_{PSIS2}x_{PSIS2} + m_{PoB1}x_{PoB1} + m_{PoB2}x_{PoB2} + m_{T1}x_{T1} + m_{T2}x_{T2} + m_{F1}x_{F1} + m_{F2}x_{F2}}{M_{total}} \quad [26]$$

$$COM_Y = \frac{m_{IT1}y_{IT1} + m_{IT2}y_{IT2} + m_{PSIS1}y_{PSIS1} + m_{PSIS2}y_{PSIS2} + m_{PoB1}y_{PoB1} + m_{PoB2}y_{PoB2} + m_{T1}y_{T1} + m_{T2}y_{T2} + m_{F1}y_{F1} + m_{F2}y_{F2}}{M_{total}} \quad [27]$$

$$COM_z = \frac{m_{IT1}z_{IT1} + m_{IT2}z_{IT2} + m_{PSIS1}z_{PSIS1} + m_{PSIS2}z_{PSIS2} + m_{PoB1}z_{PoB1} + m_{PoB2}z_{PoB2} + m_{T1}z_{T1} + m_{T2}z_{T2} + m_{F1}z_{F1} + m_{F2}z_{F2}}{M_{total}} \quad [28]$$

$$M_{Total} = \frac{User's\ Weight}{32.2\ ft/s^2} \quad [29]$$

$$PoB = Peak\ of\ Back$$
$$T = Thigh$$
$$F = Foot$$
$$1 = left\ sensor$$
$$2 = right\ sensor$$
$$*m = mass\ with\ correction\ factor\ applied$$

Below are the same steps as described above, but with experimental data when using a 130lb individual with the wheelchair reclined 10 degrees:

1. The sensors were placed on the wheelchair as previously described
2. The following table displays all of the input parameters measured. As noted earlier, the Z dimension for sensors 1-4 is equal to the seat height. The Z dimension of the foot rest is equal to the footrest height

Table 3: Input Parameters

Input Parameter	Value
User Weight	130 lb.
Seat Height	22 in
Footrest Height	3 in
Backrest Height	19 in
Recline Angle	10 Degrees
X Dim: Sensor 1	5.5 in
Y Dim: Sensor 1	3.5 in
X Dim: Sensor 2	11.5 in
Y Dim: Sensor 2	3.75 in
X Dim: Sensor 3	5.5 in
Y Dim: Sensor 3	12.5 in
X Dim: Sensor 4	13.5 in
Y Dim: Sensor 4	12 in
X Dim: Sensor 5	6.5 in
Z Dim: Sensor 5	5 in
X Dim: Sensor 6	10 in
Z Dim: Sensor 6	4 in
X Dim: Sensor 7	4.5 in
Z Dim: Sensor 7	18 in
X Dim: Sensor 8	11 in
Z Dim: Sensor 8	17.5 in
X Dim: Sensor 9	7 in
Y Dim: Sensor 9	23 in
X Dim: Sensor 10	12 in
Y Dim: Sensor 10	23 in

3. With the recline angle of 10 degrees and the Y dimensions inputted for sensors 5-8, equation 18 and 19 can be used to find the Z dim sensor 5 & 6 and 7 &8, respectively. The table below shows the dimensions after solving these equations.

Table 4: Y Dimension for Sensors 5-8

Parameter	Value
Y Dim: Sensor 5	0.868241
Y Dim: Sensor 6	0.694593
Y Dim: Sensor 7	3.125667
Y Dim: Sensor 8	3.038843

4. As described previously, MATLAB records the voltage outputs from the sensors and relates the voltage to force using a calibration equation found while calibrating each sensor. The sample calculation below shows this calculation for sensor 1.

$$Output\ \ Voltage\ Reading = 4.1496\ V$$

$$F_1 = \mathrm{e}^{\frac{4}{5749}(2500\times 4.1496V - 4901)} = 45.0556\ lbs \qquad [30]$$

5. In order to have mass outputs, the acceleration of gravity (32.2 ft. /s^2), is divided out of the force. When the backrest is angled, as it is in this case, the sensors located on the backrest receive only a portion of the gravity since gravity always acts vertically downward. Using basic trigonometry, the component of gravity that acts on the sensors can be calculated. Equation 31 below shows the calculation of mass for sensor 1, located at the user's right IT.

$$m_1 = \frac{45.0556\ lbs}{32.2} = 1.3992\ slug \qquad [31]$$

6. Next, the correction factor in Table 2, needs to be applied to make up for the mass not captured by the sensor. The following equation shows this correction factor for sensor 1.

$$1.3992\ slug \times 0.815652974 = 1.1413\ slug \qquad [32]$$

Below is a table that lists all of the voltage outputs, force values, and masses (after correction factor) for each sensor.

Table 5: Voltage, Force, and Mass Values for each Sensor

Sensor	Voltage Output (V)	Force (lbs.)	Mass (slugs)
1	4.1496	45.0556	1.399242236
2	4.1398	49.8905	1.54939441
3	3.0743	5.8161	0.180624224
4	2.9521	6.1151	0.189909938
5	1.2757	0.3262	0.010130435
6	1.0899	0.3524	0.010944099
7	0.0489	0.0053	0.000164596
8	0.2395	0.9923	0.03081677
9	3.0352	4.7068	0.146173913
10	2.8788	3.2926	0.102254658

7. The last step is to solve for the total body COM using equations 26, 27, and 28. The x, y, or z component of the equations can be found in Table 3 and 4. The mass component of the equations can be found in Table 5. Those values, with the total mass of the user, the equation can be solved. The total body COM for this user, using Equation 26, 27, and 28 were calculated to be located at X=7.72 in, Y=3.49 in, and Z=19.27 in. This location is relative to the gridded cloth on the wheelchair.

Chapter 7: Conclusion

7.1 Summary

The purpose of this study was to develop a system to experimentally calculate the center of mass for wheelchair users. This goal was achieved by using force sensing resistors placed on a wheelchair's seat and backrest. By systematically placing the sensors where the COM of each body segment is located allows for all parameters needed to solve for the total COM. The system uses an Arduino and MATLAB to collect, and analyze the data. The output result from MATLAB is the x, y, and z, location of the total body COM. This system can be utilized by researchers to help understand the effects of wheelchair configuration on the COM. This information can then be used to study effectiveness of these parameters on the posture of wheelchair users.

7.2 Contributions

This study has provided greater understanding of COM measurements with seated individuals. The methodology and instrumentation has been laid out to continue work in this area. It allows for greater research to be done in understanding how COM of humans in wheelchairs is altered and affected by pelvic positioning and overall posture. It also sets a baseline for future versions of this system.

7.3 Additional Applications

Outside of the scope of this study, this system could be utilized in the design and manufacturing of wheelchairs. By understanding how a wheelchair's design affects the user's COM can prompt designers to design the wheelchair and its components to ensure proper pelvic posture is forced and that the COM is situated appropriated above it.

Additionally, this system could be used beyond wheelchairs. Ergonomics studies could utilize this study in designing chairs, couches, and other objects meant for a seated individual. For example, in order to make air travel on planes more ergonomic and comfortable, this system could be used to ensure the COM of passengers can easily be aligned above the pelvis to ensure the seats are ergonomically sufficient.

This system can help people know their COM in any seated device or object. It can help increase the comfort of the seat and help the increase the quality of the seated individual. Having the data of COM is a parameter that can allow for the continual expansion of knowledge of how gravitational forces act on the human body and how to ensure they do not negatively impact the quality of life.

7.4 Future Work

As this device is a proof of concept prototype for creating a device to experimentally measure the COM, there are changes that could provide more accurate readings. One of these changes could be the use of a sensor that functions like the pressure mats and records all forces or pressures between the person and chair. The FSR sensors are limited in the amount of mass they are able to capture. While a correction factor is added to make up for the missing mass, the study used to create correction factors is not specific to each user in the wheelchair. It is an average from a set of test subjects from the study [6]. The downside of using a pressure mat in a measurement device such as this is that the mats are primarily designed for use on the wheelchair seats and are designed to only output the results through provided software. Using Arduino and MATLAB, the FSR sensors were simple and easy to capture data to then be use mathematical operations. The provided software for the pressure mats typically prevents the use of the mats beyond their provided software. Pressure mats can be custom made for various uses, but as with most customizations, the cost is much greater. Another change that could be added to increase is adding a filter to sensor setup to remove any feedback from the Arduino altering the results.

This project is part of a larger initiative of creating a user friendly tool for medical clinicians. In addition to this experimental measurement system, a model will be developed through OpenSim, a biomechanical modeling software that creates a model with more parameters to make it more realistic. This will be adding on the project that was done concurrent with this project to create a model to theoretically measure a patient's COM in various seated positions. All together these studies can help create a tool for clinicians to use while fitting patients to streamline the process and help them capture and understand more data on the patient pertaining to effects on the body from the wheelchair.

APPENDIX A: Calibration Curves

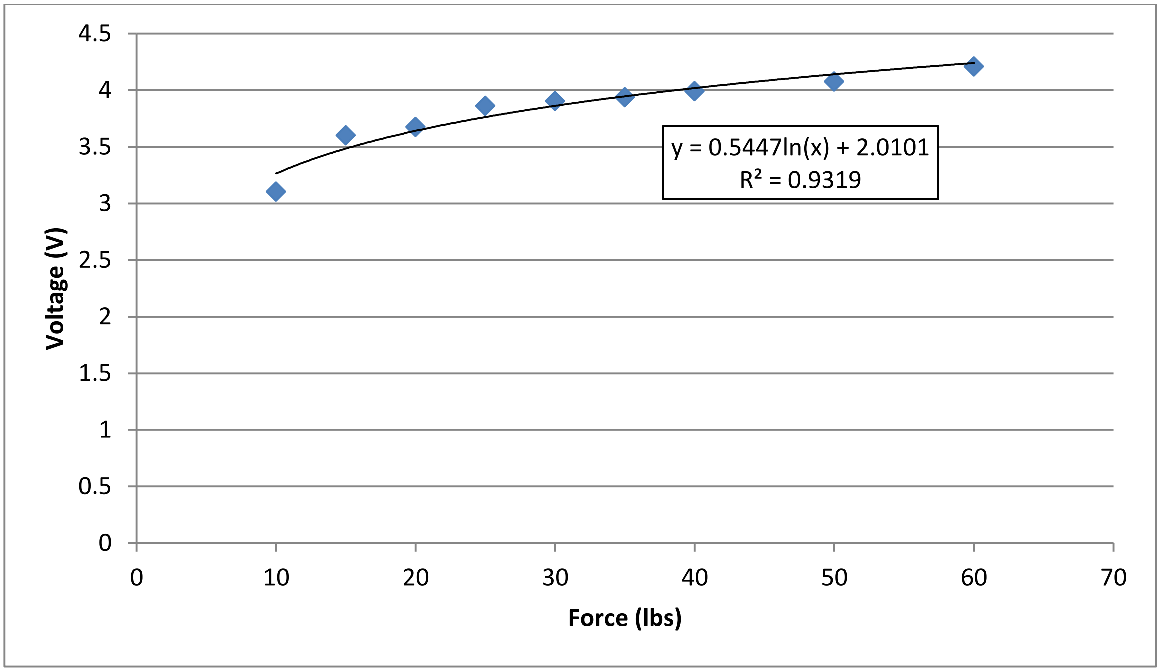

Figure 17: Sensor 2 Calibration Curve

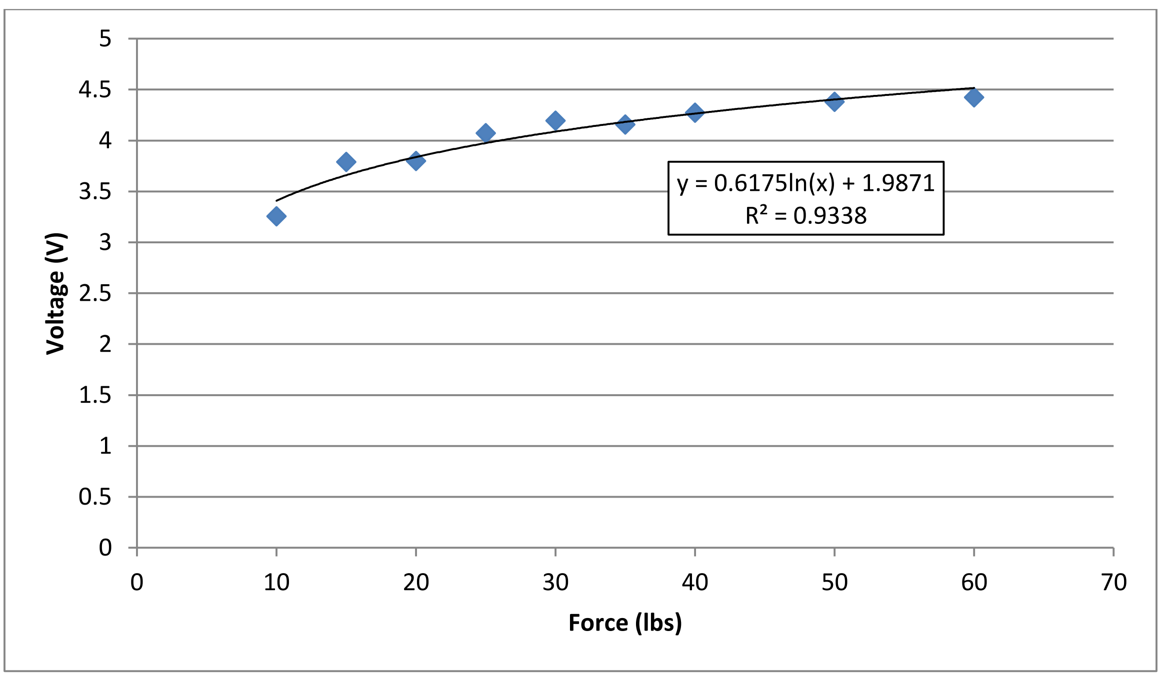

Figure 18: Sensor 3 Calibration Curve

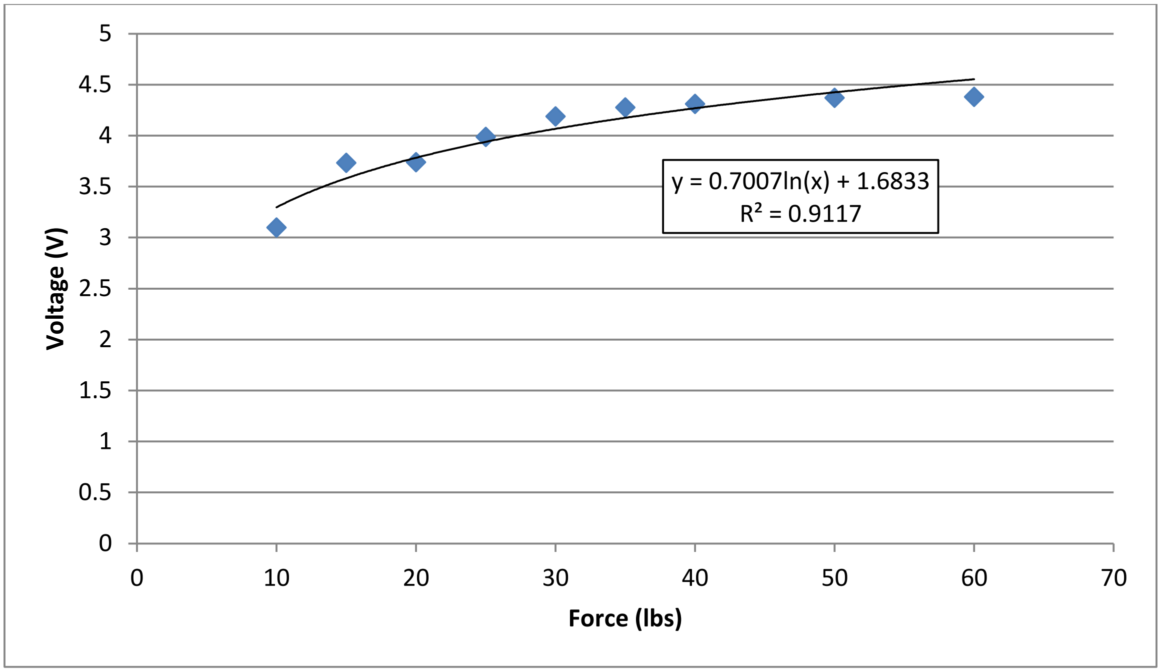

Figure 19: Sensor 4 Calibration Curve

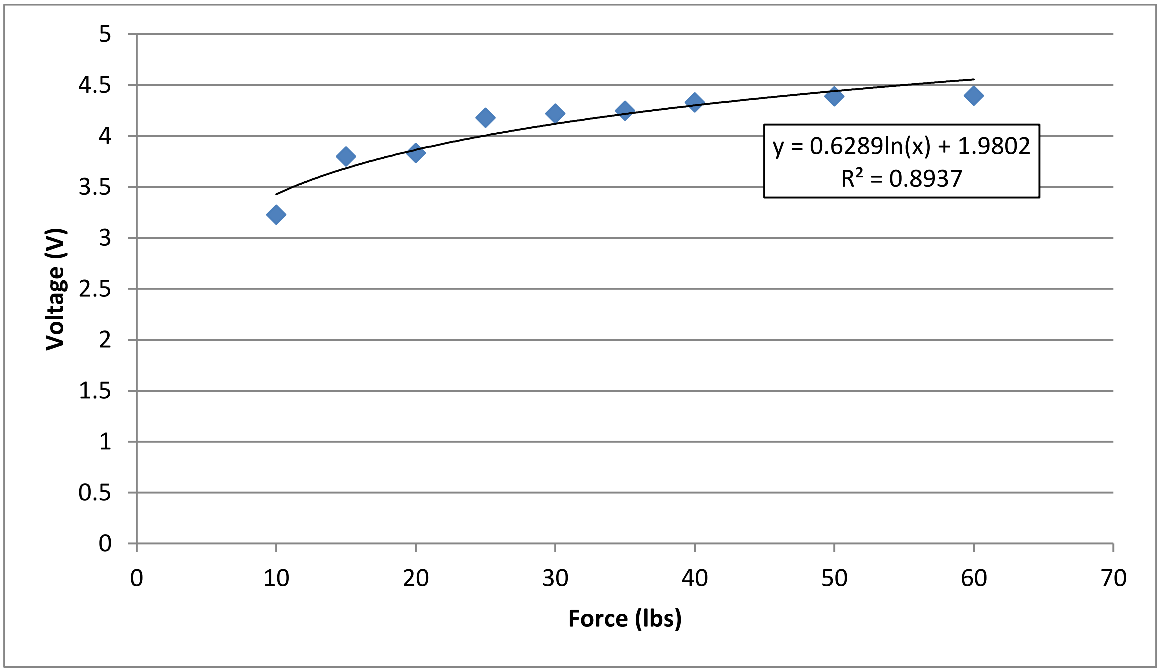

Figure 20: Sensor 5 Calibration Curve

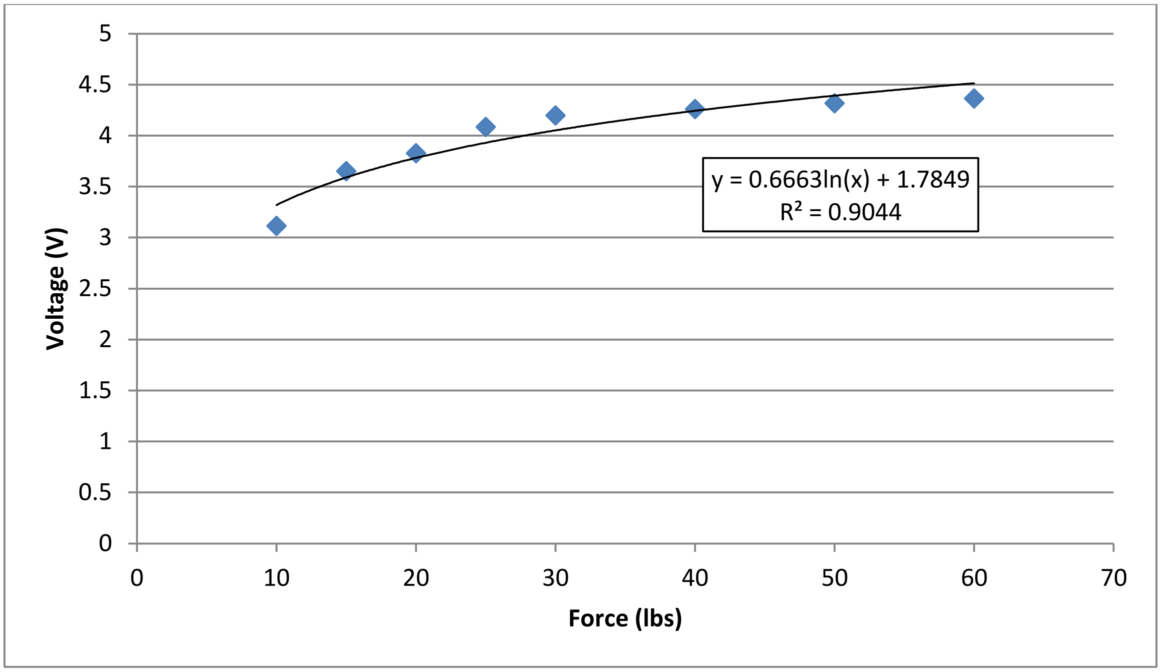

Figure 21: Sensor 6 Calibration Curve

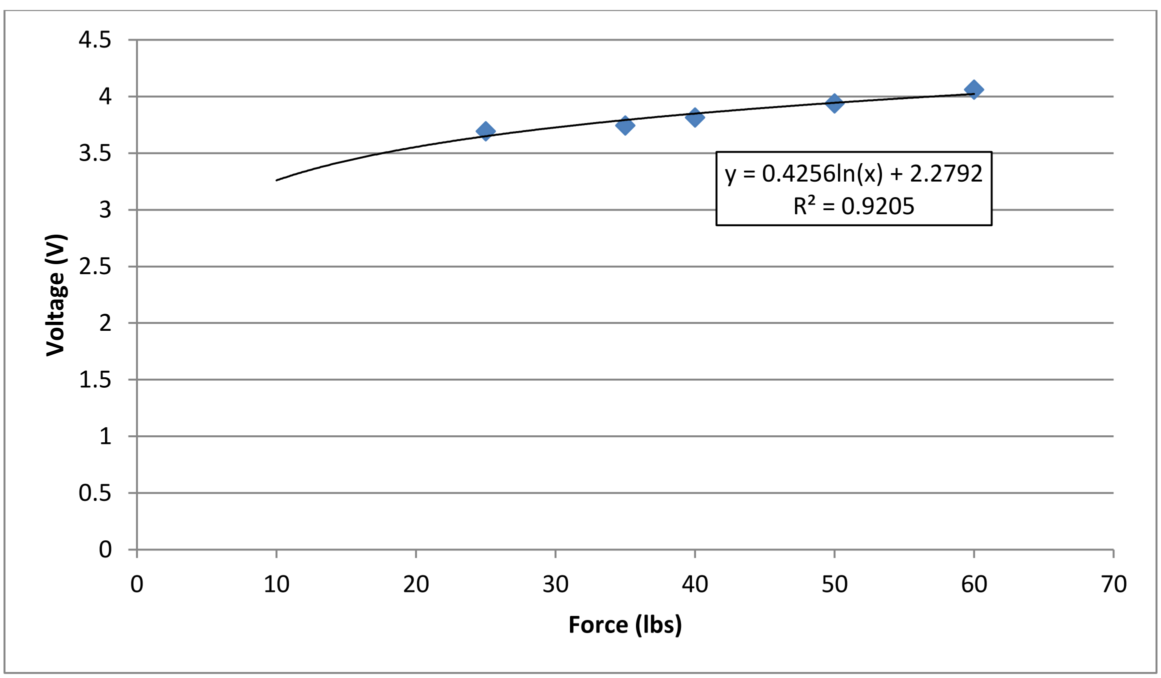

Figure 22: Sensor 7 Calibration Curve

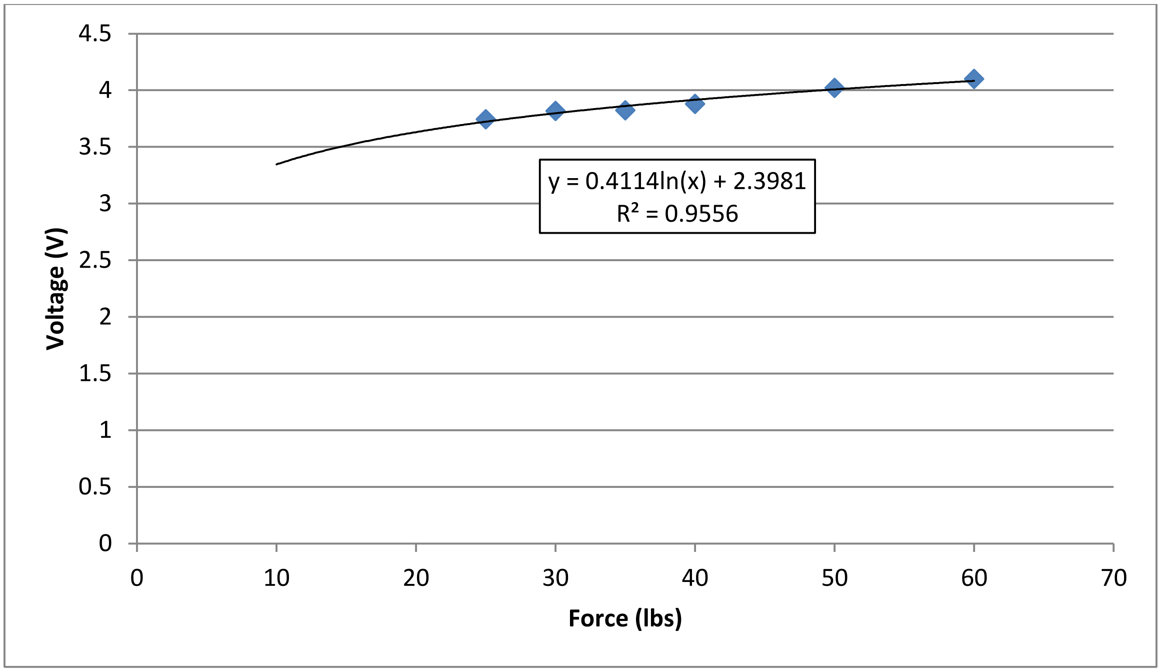

Figure 23: Sensor 8 Calibration Curve

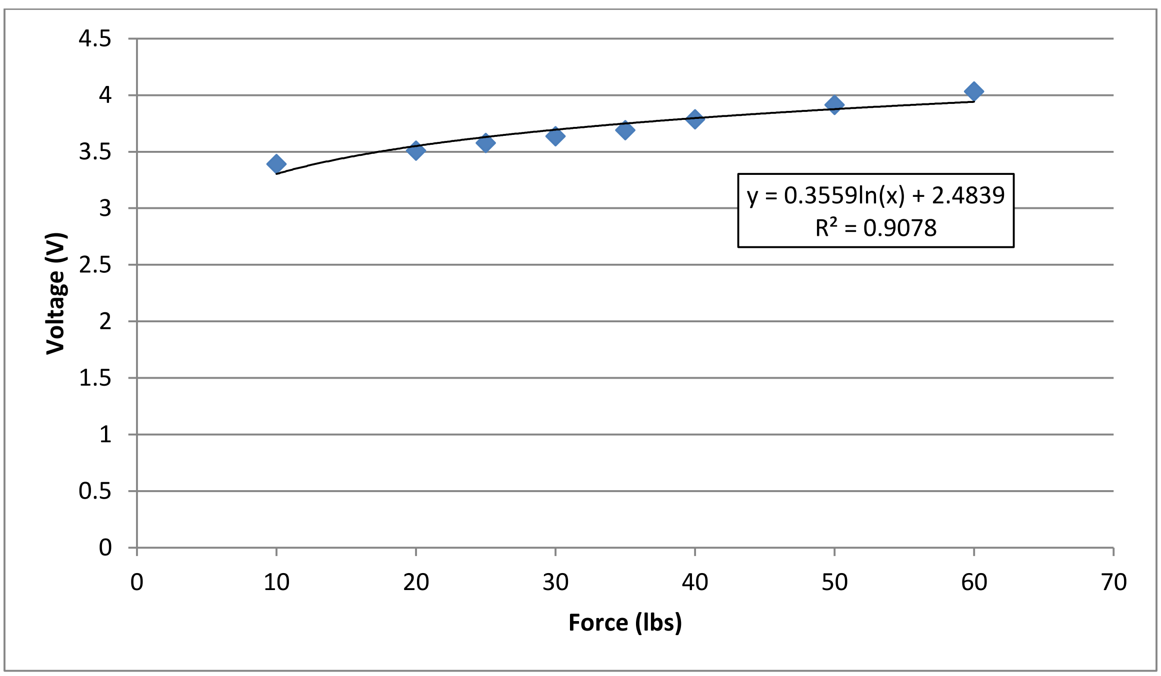

Figure 24: Sensor 9 Calibration Curve

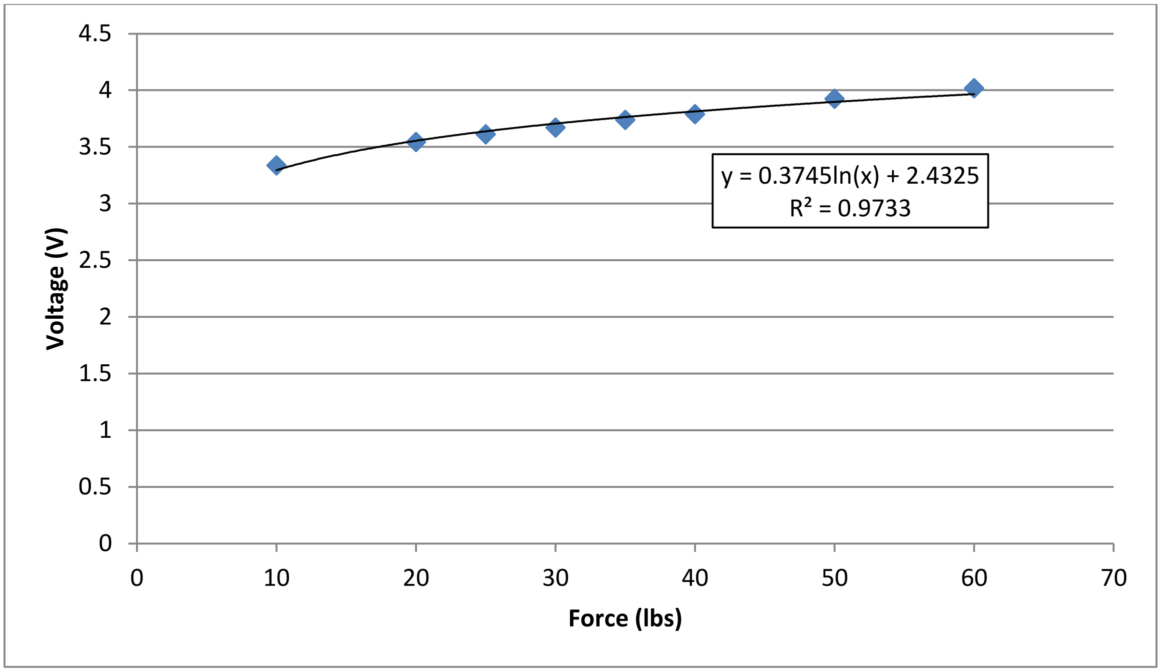

Figure 25: Sensor 10 Calibration Curve

APPENDIX B: Center of Mass MATLAB Code

```
clear all
clc

%Sensor 1&2= ITs
%Sensor 3&4=Thigh
%Sensor 5&6=PSIS
%Sensor 7&8=Peak of Back
%Sensor 9&10=Feet

%Ask for user's total weight
prompt = ['Please input the user`s mass:', '\n'];
weight=input (prompt);
g=32.2;

M=weight/g

% Ask for distances from ground for Z axis
prompt = ['Please input the wheelchair seat distance from the ground:','\n'];
zseat=input (prompt);

prompt = ['Please input the foot rest distance from the ground:','\n'];
zfoot=input (prompt);

prompt = ['Please input the distaance from the bottom of the wheelchair
backrest from the ground:','\n'];
zback=input (prompt);

% Ask for the angle of the backrest to figure out Y axis
prompt = ['Please input the wheelchair backrest angle is from the vertical
position:','\n'];
theta=input (prompt);

%Ask for x and y distances for sensors 1-4 on the wheelchair seat
prompt = ['Please input the distance in the x direction for Sensor 1:',
'\n'];
x1=input (prompt);

prompt = ['Please input the distance in the y direction for Sensor 1:',
'\n'];
y1=input (prompt);

prompt = ['Please input the distance in the x direction for Sensor 2:',
'\n'];
x2=input (prompt);

prompt = ['Please input the distance in the y direction for Sensor 2:',
'\n'];
y2=input (prompt);

prompt = ['Please input the distance in the x direction for Sensor 3:',
'\n'];
```

```
x3=input (prompt);

prompt = ['Please input the distance in the y direction for Sensor 3:',
'\n'];
y3=input (prompt);

prompt = ['Please input the distance in the x direction for Sensor 4:',
'\n'];
x4=input (prompt);

prompt = ['Please input the distance in the y direction for Sensor 4:',
'\n'];
y4=input (prompt);

%Ask for the x and z distances for sensors 5-8 on the wheelchair backrest
prompt = ['Please input the distance in the x direction for Sensor 5:',
'\n'];
x5=input (prompt);

prompt = ['Please input the distance in the z direction for Sensor 5:',
'\n'];
z5=input (prompt);

prompt = ['Please input the distance in the x direction for Sensor 6:',
'\n'];
x6=input (prompt);

prompt = ['Please input the distance in the z direction for Sensor 6:',
'\n'];
z6=input (prompt);

prompt = ['Please input the distance in the x direction for Sensor 7:',
'\n'];
x7=input (prompt);

prompt = ['Please input the distance in the z direction for Sensor 7:',
'\n'];
z7=input (prompt);

prompt = ['Please input the distance in the x direction for Sensor 8:',
'\n'];
x8=input (prompt);

prompt = ['Please input the distance in the z direction for Sensor 8:',
'\n'];
z8=input (prompt);

%Ask for the x and y distance for sensors 9 and 10 on the wheelchair
footrest.
%X and Y are measured from the origin on seat grid
prompt = ['Please input the distance in the x direction for Sensor 9:',
'\n'];
x9=input (prompt);
```

```
prompt = ['Please input the distance in the y direction for Sensor 9:',
'\n'];
y9=input (prompt);

prompt = ['Please input the distance in the x direction for Sensor 10:',
'\n'];
x10=input (prompt);

prompt = ['Please input the distance in the y direction for Sensor 10:',
'\n'];
y10=input (prompt);

%Calculate y distance of backrest sensors when reclined uses trig from angle
of seat recline
y5=z5*cos(90-theta);
y6=z6*cos(90-theta);
y7=z7*cos(90-theta);
y8=z8*cos(90-theta);

%Calculate z distance
z1=zseat;                          %measured distance from the ground to seat
z2=zseat;
z3=zseat;
z4=zseat;

z5=z5+zback;                       %measured from the grid and distance from
bottom of backrest to ground
z6=z6+zback;
z7=z7+zback;
z8=z8+zback;

z9=zfoot;                          %measured distance from the ground to the
footrest
z10=zfoot;

% Set up Arduinos
a=arduino('COM3');
b=arduino('COM4');

% Read voltages from Arduino
V1=readVoltage(a,'A0')                  %Right IT
V2=readVoltage(a,'A1')                  %Left IT
V3=readVoltage(a,'A2')                  %Right Thigh
V4=readVoltage(a,'A3')                  %Left Thigh
V5=readVoltage(a,'A4')                  %Right PSIS
V6=readVoltage(a,'A5')                  %Left PSIS
V7=readVoltage(b,'A0')                  %Right shoulder Blade
V8=readVoltage(b,'A1')                  %Left Shoulder Blade
V9=readVoltage(b,'A2')                  %Right Foot
V10=readVoltage(b,'A3')                 %Left Food

% INSERT RELATION FROM VOLTAGE TO FORCE
% Equations are found from calibrating each sensor

F1=exp(4*(2500*V1-4901)/5749)
```

```
F2=exp((10000*V2-20101)/5447)
F3=exp(400*V3/247-19871/6175)
F4=exp((10000*V4-16833)/7007)
F5=exp(2*(5000*V5-9901)/6289)
F6=exp((10000*V6-17849)/6663)
F7=exp(625*V7/266-407/76)
F8=exp((100000*V8-23981)/4114)
F9=exp((10000*V9-24839)/3559)
F10=exp(5/749*(400*V10-973))

%different calibration equation per sensor

m1=F1/g        %Sensors 1,2,3,4,9,10 are on a flat, horizontal surface (seat
or footrest)meaning gravity is acting perpendicular to sensor
m2=F2/g
m3=F3/g
m4=F4/g
m9=F9/g
m10=F10/g

%calculating the angle that gravity acts on reclined backrest

phi=180-theta;
angleg=g*cos(phi);

m5=F5/angleg
m6=F6/angleg
m7=F5/angleg
m8=F6/angleg



%MASS CORRECTION
%correction factor for the mass not captured by the sensors
%Head, arms, and trunk

%Correction factors for each sensor were determined from another test to
%make up for the mass not captured from the sensors
C1=0.815652974;
C2=1.081579;
C3=0.736368349;
C4=0.656482263;
C5=15.94979526;
C6=8.622839555;
C7=5.276004837;
C8=3.457355835;
C9=0.1888442211;
C10=0.154168722;

%Correction factor is multiplied by masses recorded
m1=m1*C1
m2=m2*C2
m3=m3*C3
m4=m4*C4
m5=m5*C5
m6=m6*C6
m7=m7*C7
```

```
m8=m8*C8
m9=m9*C9
m10=m10*C10

%CENTER OF MASS
%The previous data is compiled into the COM equations and outputs the
%coordinates of the final body COM

% x direction
COMx=(m1*x1+m2*x2+m3*x3+m4*x4+m5*x5+m6*x6+m7*x7+m8*x8+m9*x9+m10*x10)/M;

% y direction
COMy=(m1*y1+m2*y2+m3*y3+m4*y4+m5*y5+m6*y6+m7*y7+m8*y8+m9*y9+m10*y10)/M;

% z direction
COMz=(m1*z1+m2*z2+m3*z3+m4*z4+m5*z5+m6*z6+m7*z7+m8*z8+m9*z9+m10*z10)/M;

fprintf('The total center of mass is located at (%.2f, %.2f, %.2f)',COMx,
COMy, COMz);
```

APPENDIX C: Correction Factor HAT MATLAB Code

```
clear
clc

% Set up Arduino
a=arduino('COM3');

% Read voltages from Arduino
V1=readVoltage(a,'A0');                    %Yellow Left IT
V2=readVoltage(a,'A1');                    %Red Right IT
V3=readVoltage(a,'A2');                    %Green Left PSIS
V4=readVoltage(a,'A3');                    %Black Right shoulder
V5=readVoltage(a,'A4');                    %Blue Right PSIS
V6=readVoltage(a,'A5');                    %White Left shoulder

F1=exp(4*(2500*V1-4901)/5749)
F2=exp((10000*V2-20101)/5447)
F3=exp(400*V3/247-19871/6175)
F4=exp((10000*V4-16833)/7007)
F5=exp(2*(5000*V5-9901)/6289)
F6=exp((10000*V6-17849)/6663)

W=125;

Angle=10;

F1t=(0.142*W*cosd(Angle)+0.181*W*sind(Angle)+0.355*W*sind(Angle))/2
F2t=V1t
F3t=(0.355*W*cosd(Angle))/2
F4t=(0.181*W*cosd(Angle))/2
F5t=V3t
F6t=V4t

C1=F1t/F1
C2=F2t/F2
C3=F3t/F3
C4=F4t/F4
C5=F5t/F5
C6=F6t/F5
```

APPENDIX D: Correction Factor Thigh MATLAB Code

```
clear
clc

% Set up Arduino
a=arduino('COM3');

% Read voltages from Arduino
V1=readVoltage(a,'A0');                    %Yellow Left Thigh
V2=readVoltage(a,'A1');                    %Red Right Thigh

F1=exp(4*(2500*V1-4901)/5749)
F2=exp((10000*V2-20101)/5447)

W=125;

F1t=.1*W/2
F2t=F1t

C1=F1t/F1
C2=F2t/F2
```

APPENDIX E: Correction Factor Foot-Leg MATLAB Code

```
clear
clc

% Set up Arduino
a=arduino('COM3');

% Read voltages from Arduino
V1=readVoltage(a,'A0');                    %Yellow Left foot
V2=readVoltage(a,'A1');                    %Red Right foot

F1=exp(4*(2500*V1-4901)/5749)-1
F2=exp((10000*V2-20101)/5447)-1

W=125;

F1t=.06*W/2
F2t=F1t

C1=F1t/F1
C2=F2t/F2
```

References

[1] Y.-H. Kwon, "Center of Mass Lab," Oregon State University, [Online]. Available: http://oregonstate.edu/instruct/exss323/CM_Lab/Center%20of%20Mass.htm. [Accessed 06 April 2016].

[2] E. Trefler, D. A. Hobson, S. J. Taylor, L. C. Monahan and C. G. Shaw, Seating and Mobility For Persons with Physical Disabilities, Memphis: Therapy Skill Builders, 1993.

[3] J. M. Bach and K. Waugh, "Biomechanics and Its Application to Seating," Assistive Technology Partners, Univ Colorado, Denver, CO, 2015.

[4] E. Dalton, "Forward Head Posture and the 42 Pound Head," Dalton Myoskeletal, 2016. [Online]. Available: http://erikdalton.com/forward-heads-funky-necks/. [Accessed 12 April 2016].

[5] A. Kapandji, The Physiology of the Joints, Edinburgh: Churchill Livingstone, 1974.

[6] D. A. Winter, Biomechanics and Motor Control of Human Movement, Fourth Edition, Waterloo, Ontario: John Wiley & Sons, Inc., 2009.

[7] "Square Force - Sensitve Resistor (FSR) - Interlink 406," Adafruit, [Online]. Available: https://www.adafruit.com/products/1075. [Accessed 28 March 2016].

[8] Interlink Electronics, "FSR Force Sensing Resistor Integration Guide and Evaluation Parts Catalog," [Online]. Available: https://www.sparkfun.com/datasheets/Sensors/Pressure/fsrguide.pdf. [Accessed 28 March 2016].

[9] Arduino, "What is Arduino?," Arduino, 2016. [Online]. Available: http://www.arduino.cc/en/Guide/Introduction. [Accessed 28 March 2016].

[10] J. Blum, Exploring Arduino Tools and Techniques for engineering wizardry, Indianapolis: John Wiley & Sons, Inc., 2013.

[11] "MATLAB Product Description," The Mathworks, Inc., 2016. [Online]. Available: http://www.mathworks.com/help/matlab/learn_matlab/product-description.html. [Accessed 28 March 2016].

[12] "MATLAB Support Package for Arduino Hardware," The Mathworks, Inc., 2016. [Online]. Available: http://www.mathworks.com/help/supportpkg/arduinoio/index.html?searchHighlight=arduino. [Accessed 28 March 2016].